class13

```
library(DESeq2)
```

```
Loading required package: S4Vectors
```

```
Loading required package: stats4
```

```
Loading required package: BiocGenerics
```

```

Attaching package: 'BiocGenerics'
```

```
The following objects are masked from 'package:stats':

    IQR, mad, sd, var, xtabs
```

```
The following objects are masked from 'package:base':

    anyDuplicated, append, as.data.frame, basename, cbind, colnames,
    dirname, do.call, duplicated, eval, evalq, Filter, Find, get, grep,
    grepl, intersect, is.unsorted, lapply, Map, mapply, match, mget,
    order, paste, pmax, pmax.int, pmin, pmin.int, Position, rank,
    rbind, Reduce, rownames, sapply, setdiff, sort, table, tapply,
    union, unique, unsplit, which.max, which.min
```

```

Attaching package: 'S4Vectors'
```

```
The following objects are masked from 'package:base':

    expand.grid, I, unname

Loading required package: IRanges

Loading required package: GenomicRanges

Loading required package: GenomeInfoDb

Loading required package: SummarizedExperiment

Loading required package: MatrixGenerics

Loading required package: matrixStats


Attaching package: 'MatrixGenerics'

The following objects are masked from 'package:matrixStats':

    colAlls, colAnyNAs, colAnys, colAvgsPerRowSet, colCollapse,
    colCounts, colCummaxs, colCummins, colCumprods, colCumsums,
    colDiffs, colIQRDiffs, colIQRs, colLogSumExps, colMadDiffs,
    colMads, colMaxs, colMeans2, colMedians, colMins, colOrderStats,
    colProds, colQuantiles, colRanges, colRanks, colSdDiffs, colSds,
    colSums2, colTabulates, colVarDiffs, colVars, colWeightedMads,
    colWeightedMeans, colWeightedMedians, colWeightedSds,
    colWeightedVars, rowAlls, rowAnyNAs, rowAnys, rowAvgsPerColSet,
    rowCollapse, rowCounts, rowCummaxs, rowCummins, rowCumprods,
    rowCumsums, rowDiffs, rowIQRDiffs, rowIQRs, rowLogSumExps,
    rowMadDiffs, rowMads, rowMaxs, rowMeans2, rowMedians, rowMins,
    rowOrderStats, rowProds, rowQuantiles, rowRanges, rowRanks,
    rowSdDiffs, rowSds, rowSums2, rowTabulates, rowVarDiffs, rowVars,
    rowWeightedMads, rowWeightedMeans, rowWeightedMedians,
    rowWeightedSds, rowWeightedVars

Loading required package: Biobase
```

```
Welcome to Bioconductor

    Vignettes contain introductory material; view with
    'browseVignettes()'. To cite Bioconductor, see
    'citation("Biobase")', and for packages 'citation("pkgname")'.


Attaching package: 'Biobase'

The following object is masked from 'package:MatrixGenerics':

    rowMedians

The following objects are masked from 'package:matrixStats':

    anyMissing, rowMedians
```

```
colData <- read.csv("GSE37704_metadata.csv",row.names = 1)
colData
```

```
             condition
SRR493366 control_sirna
SRR493367 control_sirna
SRR493368 control_sirna
SRR493369      hoxa1_kd
SRR493370      hoxa1_kd
SRR493371      hoxa1_kd
```

```
countData <- read.csv("GSE37704_featurecounts.csv",row.names = 1)
head(countData)
```

```
                length SRR493366 SRR493367 SRR493368 SRR493369 SRR493370
ENSG00000186092    918         0         0         0         0         0
ENSG00000279928    718         0         0         0         0         0
ENSG00000279457   1982        23        28        29        29        28
ENSG00000278566    939         0         0         0         0         0
ENSG00000273547    939         0         0         0         0         0
ENSG00000187634   3214       124       123       205       207       212
```

```
                SRR493371
ENSG00000186092         0
ENSG00000279928         0
ENSG00000279457        46
ENSG00000278566         0
ENSG00000273547         0
ENSG00000187634       258
```

Q. Complete the code below to remove the troublesome first column from countData

```
countData<- countData[,-1]
head(countData)
```

```
                SRR493366 SRR493367 SRR493368 SRR493369 SRR493370 SRR493371
ENSG00000186092         0         0         0         0         0         0
ENSG00000279928         0         0         0         0         0         0
ENSG00000279457        23        28        29        29        28        46
ENSG00000278566         0         0         0         0         0         0
ENSG00000273547         0         0         0         0         0         0
ENSG00000187634       124       123       205       207       212       258
```

```
#countData <- as.matrix(countData[,-1])
#head(countData)
```

```
all(rownames(colData) == colnames(countData))
```

```
[1] TRUE
```

Q. Complete the code below to filter countData to exclude genes (i.e. rows) where we have 0 read count across all samples (i.e. columns).

```
keep.inds <- rowSums(countData) != 0
```

```
counts <- countData[keep.inds, ]
head(counts)
```

```
                SRR493366 SRR493367 SRR493368 SRR493369 SRR493370 SRR493371
ENSG00000279457        23        28        29        29        28        46
ENSG00000187634       124       123       205       207       212       258
ENSG00000188976      1637      1831      2383      1226      1326      1504
ENSG00000187961       120       153       180       236       255       357
ENSG00000187583        24        48        65        44        48        64
ENSG00000187642         4         9        16        14        16        16
```

```
nrow(counts)
```

```
[1] 15975
```

```
library(DESeq2)
```

```
dds <- DESeqDataSetFromMatrix(countData=countData,colData=colData,design=~condition)
```

```
Warning in DESeqDataSet(se, design = design, ignoreRank): some variables in
design formula are characters, converting to factors
```

```
dds <- DESeq(dds)
```

```
estimating size factors
```

```
estimating dispersions
```

```
gene-wise dispersion estimates
```

```
mean-dispersion relationship
```

```
final dispersion estimates
```

```
fitting model and testing
```

```
res <- results(dds)
head(res)
```

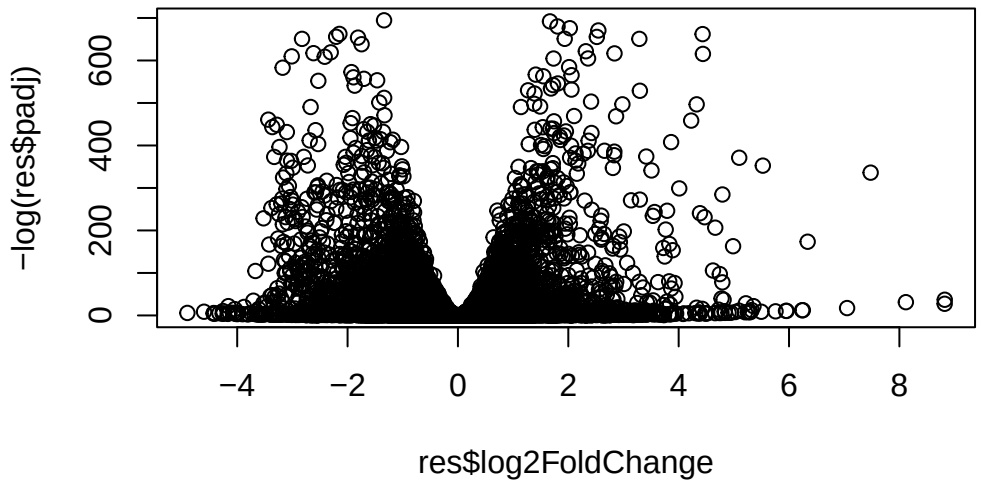
```
log2 fold change (MLE): condition hoxa1 kd vs control sirna
Wald test p-value: condition hoxa1 kd vs control sirna
DataFrame with 6 rows and 6 columns
                 baseMean log2FoldChange     lfcSE      stat     pvalue
                <numeric>      <numeric> <numeric> <numeric>  <numeric>
ENSG00000186092    0.0000             NA        NA        NA         NA
ENSG00000279928    0.0000             NA        NA        NA         NA
ENSG00000279457   29.9136       0.179257  0.324822  0.551863 0.58104205
ENSG00000278566    0.0000             NA        NA        NA         NA
ENSG00000273547    0.0000             NA        NA        NA         NA
ENSG00000187634  183.2296       0.426457  0.140266  3.040350 0.00236304
                     padj
                <numeric>
ENSG00000186092         NA
ENSG00000279928         NA
ENSG00000279457 0.68707978
ENSG00000278566         NA
ENSG00000273547         NA
ENSG00000187634 0.00516278
```

Q. Call the summary() function on your results to get a sense of how many genes are up or down-regulated at the default 0.1 p-value cutoff.

```
summary(res)
```

```
out of 15975 with nonzero total read count
adjusted p-value < 0.1
LFC > 0 (up)       : 4349, 27%
LFC < 0 (down)     : 4393, 27%
outliers [1]       : 0, 0%
low counts [2]     : 1221, 7.6%
(mean count < 0)
[1] see 'cooksCutoff' argument of ?results
[2] see 'independentFiltering' argument of ?results
```

```
plot( res$log2FoldChange, -log(res$padj) )
```

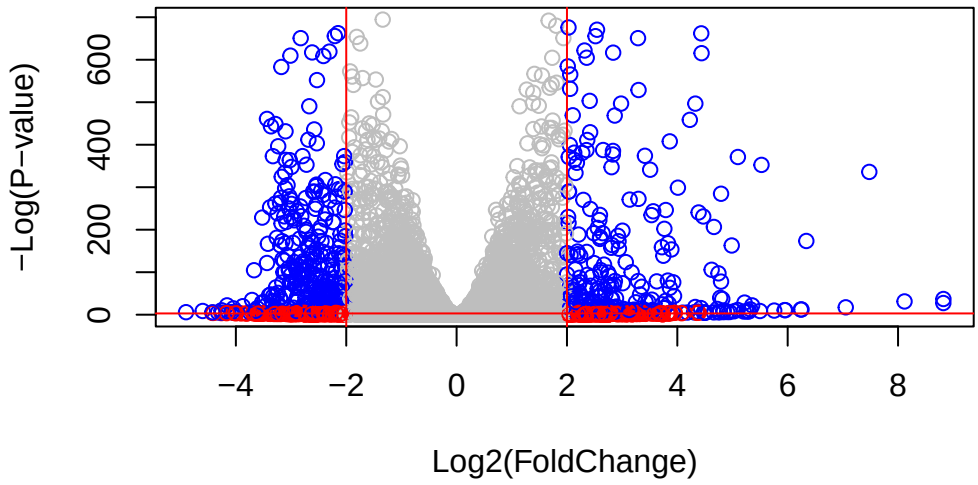


Q. Improve this plot by completing the below code, which adds color and axis labels

```
mycols <- rep("grey", nrow(res) )
mycols[ abs(res$log2FoldChange) > 2 ] <- "red"
inds <- (res$padj < 0.01) & (abs(res$log2FoldChange) > 2 )
mycols[ inds ] <- "blue"

plot( res$log2FoldChange, -log(res$padj), col=mycols, xlab="Log2(FoldChange)", ylab="-Log(

abline(v=c(-2,2), col="red")
abline(h=-log(0.05), col="red")
```



Q. Use the mapIDs() function multiple times to add SYMBOL, ENTREZID and GENENAME annotation to our results by completing the code below.

```
library("AnnotationDbi")
library("org.Hs.eg.db")
```

```
columns(org.Hs.eg.db)
```

```
 [1] "ACCNUM"       "ALIAS"        "ENSEMBL"      "ENSEMBLPROT"  "ENSEMBLTRANS"
 [6] "ENTREZID"     "ENZYME"       "EVIDENCE"     "EVIDENCEALL"  "GENENAME"
[11] "GENETYPE"     "GO"           "GOALL"        "IPI"          "MAP"
[16] "OMIM"         "ONTOLOGY"     "ONTOLOGYALL"  "PATH"         "PFAM"
[21] "PMID"         "PROSITE"      "REFSEQ"       "SYMBOL"       "UCSCKG"
[26] "UNIPROT"
```

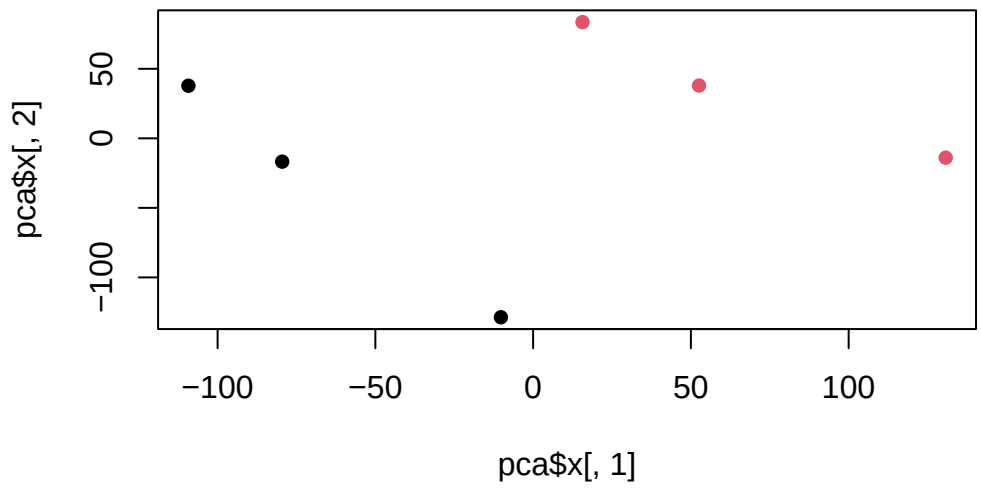
```
res$symbol <- mapIds(org.Hs.eg.db,
                     keys=row.names(counts),
                     keytype="ENSEMBL",
                     column="SYMBOL")
res$entrez <- mapIds(org.Hs.eg.db,
                     keys=row.names(counts),
                     keytype="ENSEMBL",
```

```
                      column="ENTREZID",
                      multiVals="first")
res$name <- mapIds(org.Hs.eg.db,
                      keys=row.names(counts),
                      keytype="ENSEMBL",
                      column="GENENAME")
head(counts)
```

```
pca <- prcomp(t(counts),scale=TRUE)
summary(pca)
```

```
Importance of components:
                           PC1     PC2      PC3      PC4      PC5        PC6
Standard deviation     87.7211 73.3196 32.89604 31.15094 29.18417 6.648e-13
Proportion of Variance  0.4817  0.3365  0.06774  0.06074  0.05332 0.000e+00
Cumulative Proportion   0.4817  0.8182  0.88594  0.94668  1.00000 1.000e+00
```

```
#colData
plot(pca$x[,1],pca$x[,2],col=as.factor(colData$condition),pch=16)
```




```
library(pathview)
library(gage)
library(gageData)
```

```
data(kegg.sets.hs)
data(sigmet.idx.hs)

kegg.sets.hs = kegg.sets.hs[sigmet.idx.hs]

#head(kegg.sets.hs, 3)
```

```
foldchanges <- res$log2FoldChange
names(foldchanges) <- res$entrez
head(foldchanges)
```

```
[1]        NA        NA 0.1792571        NA        NA 0.4264571
```

```
keggres = gage(foldchanges, gsets=kegg.sets.hs)
```

```
pathview(gene.data=foldchanges, pathway.id="hsa04110")
```

```
Warning: None of the genes or compounds mapped to the pathway!
Argument gene.idtype or cpd.idtype may be wrong.

'select()' returned 1:1 mapping between keys and columns

Info: Working in directory /Users/chrisz/Desktop/BGGN 213/WEEK7/Class 13

Info: Writing image file hsa04110.pathview.png
```

```
data(go.sets.hs)
data(go.subs.hs)

# Focus on Biological Process subset of GO
gobpsets <- go.sets.hs[go.subs.hs$BP]

gobpres <- gage(foldchanges, gsets=gobpsets, same.dir=TRUE)

head(gobpres$less)
```

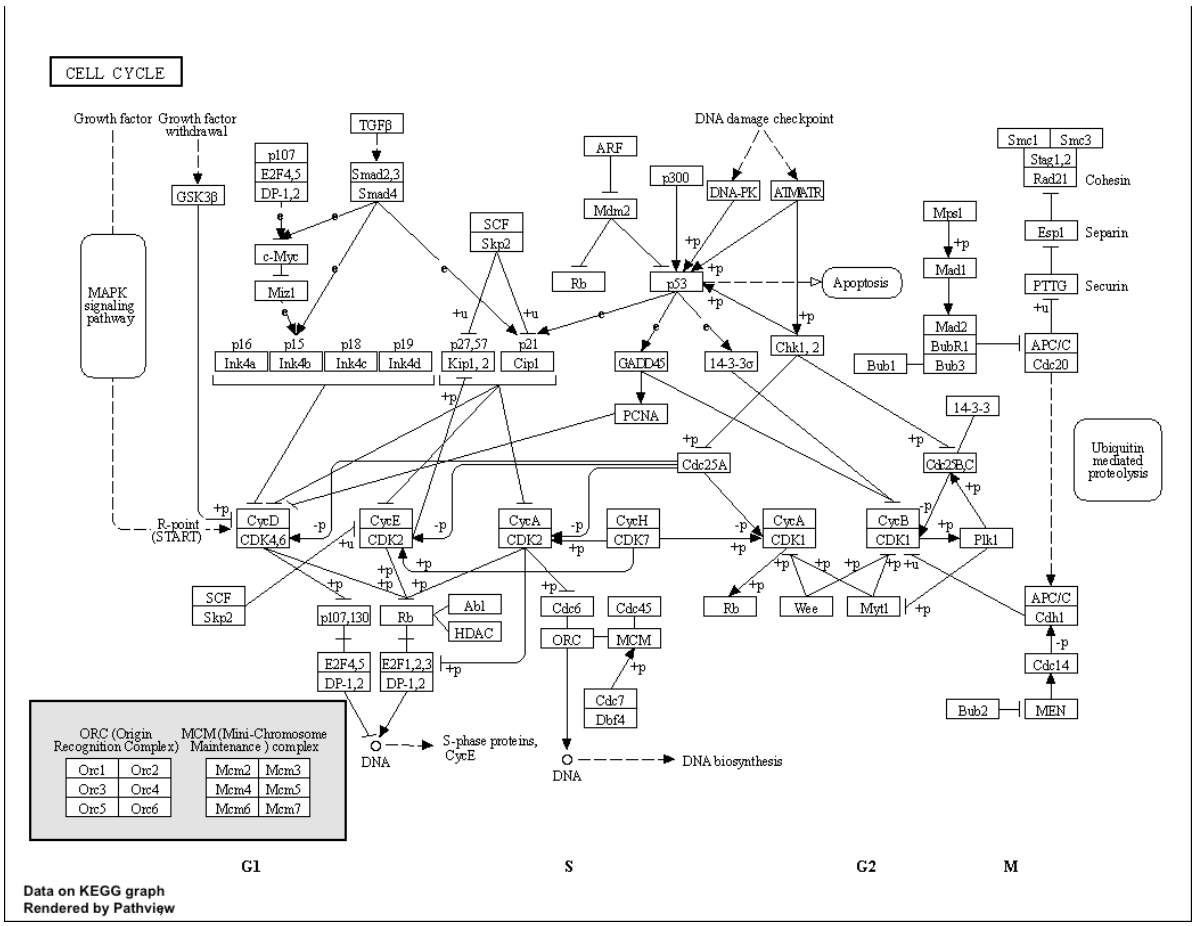


Figure 1: hsa04110 Cell cycle

```
                                          p.geomean stat.mean p.val q.val
GO:0000002 mitochondrial genome maintenance        NA       NaN    NA    NA
GO:0000003 reproduction                            NA       NaN    NA    NA
GO:0000012 single strand break repair              NA       NaN    NA    NA
GO:0000018 regulation of DNA recombination         NA       NaN    NA    NA
GO:0000019 regulation of mitotic recombination     NA       NaN    NA    NA
GO:0000022 mitotic spindle elongation              NA       NaN    NA    NA
                                          set.size exp1
GO:0000002 mitochondrial genome maintenance        0   NA
GO:0000003 reproduction                            0   NA
GO:0000012 single strand break repair              0   NA
GO:0000018 regulation of DNA recombination         0   NA
GO:0000019 regulation of mitotic recombination     0   NA
GO:0000022 mitotic spindle elongation              0   NA
```

```
#lapply(gobpres,head)
```

```
#sig_genes <- res[res$padj <= 0.05 & !is.na(res$padj), "symbol"]
#print(paste("Total number of significant genes:", length(sig_genes)))
```

```
#write.table(sig_genes, file="significant_genes.txt", row.names=FALSE, col.names=FALSE, qu
```